TCS 225

Tactical Control System (TCS)

to

Tactical Communications Interface Module (TCIM)

Interface Design Description (IDD)

Prepared for:
Program Executive Officer, Cruise Missiles Project
and Unmanned Air Vehicles Joint Project

Prepared by:
Naval Surface Warfare Center
Dahlgren Division
Version 1.0
30 September 1997

Approved by: ______________________________

TCS Program Manager

Date: ______________________________

DRAFT

TABLE OF CONTENTS

TCS to TCIM Interface Design Description

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

1. Scope

1.1 Identification

This IDD identifies, specifies, and establishes the detailed interface requirements for the TCS to TCIM communication interface. The TCIM will permit the TCS to exchange Intelligence and Electronic Warfare Character-Oriented Message Catalog (IEWCOMCAT), Army Command and Control System (ACCS), United States Message Text Format (USMTF), and Variable Message Format (VMF) messages with the All Source Analysis System (ASAS) and Advanced Field Artillery Tactical Data System (AFATDS). Given the low data rate nature of the combined Mobile Subscriber Equipment (MSE) and Single Channel Ground and Airborne Radio System (SINCGARS) communication environment, there is no plan to exchange National Imagery Transmission Format (NITF) files through the TCIM interface.

This IDD specifies interface requirements levied on the TCS, and does not impose any requirements on the Command, Control, Communications, Computers and Intelligence (C^4I) systems addressed in this document. This IDD further specifies the methods to be used to ensure that each system interface requirement has been met. This IDD is published in accordance with Data Item Description (DID) DI-IPSC-81436, dated 941205, and modified to incorporate the qualification provisions section that is traditionally found in the Interface Requirements Specification (IRS). This IDD will be revised at the conclusion of the Program Definition and Risk Reduction period of the TCS program and will be re-issued in final form to be used during the follow-on TCS Engineering and Manufacturing Development period.

1.2 System Overview

The purpose of the TCS is to provide the military services with a single command, control, data receipt, data processing, data export, and data dissemination capability that is interoperable with the family of all present and future unmanned aerial vehicles (UAV) and designated C^4I systems.

These UAVs shall include the Tactical Unmanned Aerial Vehicle (TUAV) and the Medium Altitude Endurance (MAE) UAV (henceforth referred to as Outrider and Predator, respectively), with their associated payloads. The TCS will also be capable of receiving and processing information from High Altitude Endurance (HAE) UAVs and their associated payloads, as well as being capable of providing interoperability with future UAVs and payloads.

1.2.1 TCS Program, Phases, and UAV Interaction

The Unmanned Aerial Vehicle Joint Project Office (UAV JPO) has undertaken development of a TCS for UAVs. Design and development of the TCS will be conducted in two phases. Phase 1 is defined as the Program Definition and Risk Reduction phase, and Phase 2 is defined as the Engineering and Manufacturing Development phase in accordance with Department Of Defense Instruction (DoDI) - 5000.2R. During Phase 2, TCS Low Rate Initial Production (LRIP) will commence. Phase 1 will be a 24 month period and will demonstrate Level 1 through Level 5 interaction (as defined below) in an Incremental and Evolutionary strategy as described in accordance with MIL-STD-498. The five discrete levels of multiple UAV interaction to be provided by the TCS are:

Level 1: receipt and transmission of secondary imagery and/or data
Level 2: direct receipt of imagery and/or data
Level 3: control of the UAV payload in addition to direct receipt of imagery/data
Level 4: control of the UAV, less launch and recovery, plus all the functions of level three
Level 5: capability to have full function and control of the UAV from takeoff to landing

1.2.2 Tactical Control System

The TCS is the software, software-related hardware and the extra ground support hardware necessary for the control of the TUAV, the MAE UAV, and future UAVs. The TCS will also provide connectivity to specific C^4I systems. The TCS will have the objective capability of receiving HAE UAV payload information. Although developed as a total package, the TCS will be scaleable to meet the user's requirements for deployment. The TCS will provide a common Human-Computer Interface (HCI) for tactical airborne platforms to simplify user operations and training, and to facilitate seamless integration into the Services' joint C^4I infrastructure across all levels of interaction.

1.2.2.1 Software

The major focus of the TCS program is software. The software will provide the UAV operator the necessary tools for computer related communications, mission tasking, mission planning, mission execution, data receipt, data processing, limited data exploitation, and data dissemination. The software will provide a high resolution computer generated graphics user interface that enables a UAV operator trained on one system to control different types of UAVs or UAV payloads with a minimum of additional training. The TCS will operate in an open architecture and be capable of being hosted on computers that are typically supported by the using Service. Software developed will be Defense Information Infrastructure / Common Operating Environment (DII/COE) compliant, non-proprietary, and the architectural standard for all future tactical UAVs. To the extent possible, the TCS will use standard DoD software components to achieve commonality. The TCS will provide software portability, scaleable functionality, and support for operational configurations tailored to the users' needs.

1.2.2.2 Hardware

To the extent possible, the TCS will use standard Department of Defense (DoD) components in order to achieve commonality. The TCS will use the computing hardware specified by the service-specific procurement contracts. The individual armed services will identify TCS computing hardware, the desired level of TCS functionality, the battlefield C^4I connectivity, and the particular type of air vehicle and payloads to be operated depending upon the deployment concept and area of operations. The TCS hardware must be capable of being scaled or modularized to meet varying Service needs. The TCS hardware will permit long range communications from one TCS to another, data storage expansion, access to other computers to share in processing capability, and multiple external peripherals.

1.2.2.3 System Compliance

The TCS will be developed in compliance with the following military and commercial computing systems architecture, communications processing, and imagery architecture standards:

a) Department of Defense (DoD) Joint Technical Architecture (JTA), including but not limited to:

1. Variable Message Format (VMF) and Joint Message Format (JMF)
2. National Imagery Transmission Format (NITF)

b) Defense Information Infrastructure (DII) Common Operating Environment (COE)

c) Computer Open Systems Implementation Program (COSIP)

d) Common Imagery Ground/Surface System (CIGSS) Segment of Distributed Common Ground Station (DCGS)

1.2.2.4 Integration with Joint C4I Systems

The TCS will be capable of entering DII-COE compliant networks, and TCS integration with C^4I systems will be accomplished through development of interfaces that permit information exchange between the TCS and specified C^4I systems. Network interoperability will include but not be limited to:

Army Mission Planning System (AMPS)

Advanced Tomahawk Weapons Control System (ATWCS)

Advanced Field Artillery Tactical Data System (AFATDS)

Air Force Mission Support System (AFMSS)

All Source Analysis System (ASAS)

Automated Target Hand-off System (ATHS)

Closed Circuit Television (CCTV)

Common Operational Modeling, Planning, and Simulation Strategy (COMPASS)

Contingency Airborne Reconnaissance System (CARS)

Enhanced Tactical Radar Correlator (ETRAC)

Guardrail Common Sensor/Aerial Common Sensor (ACS) Integrated Processing Facility (IPF)

Intelligence Analysis System (IAS)

Joint Deployable Intelligence Support System (JDISS)

Joint Maritime Command Information System (JMCIS)

Joint Service Imagery Processing System - Air Force (JSIPS-AF)

Joint Service Imagery Processing System - Navy (JSIPS-N)

Joint Surveillance Target Attack Radar System (JSTARS) Ground Station Module/Common Ground Station (GSM/CGS)

Modernized Imagery Exploitation System (MIES)

Tactical Aircraft Mission Planning System (TAMPS)

Tactical Exploitation Group (TEG)

Tactical Exploitation System (TES)

Theater Battle Management Core System (TBMCS)

TROJAN Special Purpose Integrated Remote Intelligence Terminal (SPIRIT) II

The TCS will export and disseminate UAV imagery products, tactical communication messages, as well as mission plans and target coordinates. The TCS also will receive, process and display tasking orders and operational information from Service-specific mission planning systems.

1.2.3 TCIM System Overview

The TCIM is a programmable two-channel communications device that supports multiple protocols and communication interfaces. TCS will use the TCIM to interface with MSE and/or SINCGARS for tactical communications. TCS will communicate with the TCIM through the Communication (Comm) Server Computer Software Component (CSC) of the DII COE Communications Software (CS).

1.3 Document Overview

This IDD describes the interface between the TCS and the TCIM. The information provided in this IDD applies to all TCIM models.

This document was developed using MIL-STD-498 (Data Item Description DI-IPSC-84136) as a guide, and is divided into the following sections:

Section 1 Scope: Provides identification of the systems, interfacing entities, and interfaces which are addressed in this IDD; and gives a brief overview of these systems.

Section 2 Referenced Documents: Lists all referenced documents applicable to this development effort.

Section 3 Interface Design: Identifies and describes the characteristics of the interface(s) defined in this IDD.

Section 4 Requirements Traceability and Qualification Provisions: Define the specific functions which will be performed over the TCS/TCIM interface, the applicable C4I system used, Operational Requirements Document (ORD) Requirement(s)/System /Subsystem Specification (SSS) Requirement(s) applicable to each function and the qualification method used to validate the functions of the interface.”.

Section 5 Notes: Provides background information regarding the TCIM; and a list of acronyms and abbreviations used in this IDD.

2. Referenced Documents

This section identifies the applicable documents for this IDD.

2.1 Government Documents

The following documents of the exact issue shown form part of this IDD to the extent specified herein. In the event of conflict between the documents referenced herein and the content of this IDD, the content of this IDD will be considered a superseding requirement.

2.1.1 Specifications

Data Item Description (DID) DI-IPSC-81436, December 5, 1994

Tactical Control System / Subsystem Specification (Version 1.0), 30 June1997

Software Requirements Specification (Version 1.1), 1 August1997

Tactical Control System (TCS) to Advanced Field Artillery Tactical Data System (AFATDS) Interface Design Document, DRAFT, August 6, 1997

Tactical Control System (TCS) to All Source Analysis System (ASAS) Interface Design Document, Version 1.0, September 8, 1997

2.1.2 Standards

Federal — None.

Military —

Field Manual No 24-7 Final Draft - June 1995	Army Battle Command System (ABCS) Systems Management Techniques
MIL-STD-498 5 Dec. 1994	Software Development and Documentation Standard
MIL-STD-188-220B Draft - 12 Jun 1997	Interoperability Standard for Digital Message Transfer Device Subsystems
MIL-STD-2045-47001B Draft - 12 June 1997	Interoperability Standard for Connectionless Data Transfer Application Layer Standard
MIL-STD-6040	U.S. Message Text Formatting program Description

1 January 1997	of U.S. Message Text Formatting Program
CJCSM 6120.05 1 January 1997	Manual for Tactical Command and Control Planning Guidance for Joint Operations, Joint Interface Operational procedures for Message Text Formats
JIEO Circular 9152 1 January 1997	Repository of United States Message Text Format (USMTF) Program Items for U.S. Implementation Guidance
VMF TIDP-TE Reissue 2, August 1996	Variable Message Format (VMF) Technical Interface Design Plan - Test Edition (TIDP - TE)

Other Government Agency —

DoD JTA Ver. 1.0 22 August 1996	Department of Defense (DoD) Joint Technical Architecture (JTA), Version 1.0

2.1.3 Drawings

None.

2.1.4 Other Publications

Reports —

NSWCDD/96-XX 9 Dec. 1996	Operational Concept Document for the TCS (Draft)
JROCM 011-97 Version 5.0 3 February 1997	Operational Requirements Document for UAV TCS
DII/COE I&RTS January 1997	DII COE Integration and Runtime Specification (I&RTS), Revision 3.0

Regulations — None.

Handbooks —

DCGS-Hdbk Version 1.0 31 July 1997	DCGS Acquisition Handbook

Bulletins — None.

2.2 Non-Government Documents

The following documents, of the exact issue shown, form part of this IDD to the extent specified herein. In the event of conflict between the documents referenced herein and the content of this IDD, the content of this IDD will be considered a superseding requirement.

2.2.1 Specifications

IDD-0288-S001 Interface Design Document (IDD) for the Tactical
Rev Q: May 15, 1997 Communications Interface Module (TCIM) Common Interface
Software (TCIS) to Applications Software ,
Litton Data Systems, San Diego, CA 92121

IDD/N250-92-L029-009 Interface Design Document (IDD) for the Communications Server
Rev D2: August 22, 1997 Computer Software Component (CSC) of the Common Operating
Environment (COE) Communications Software (CS),
Litton Data Systems, San Diego, CA 92121

2.2.2 Standards

None.

2.2.3 Drawings

None.

2.2.4 Other Publications

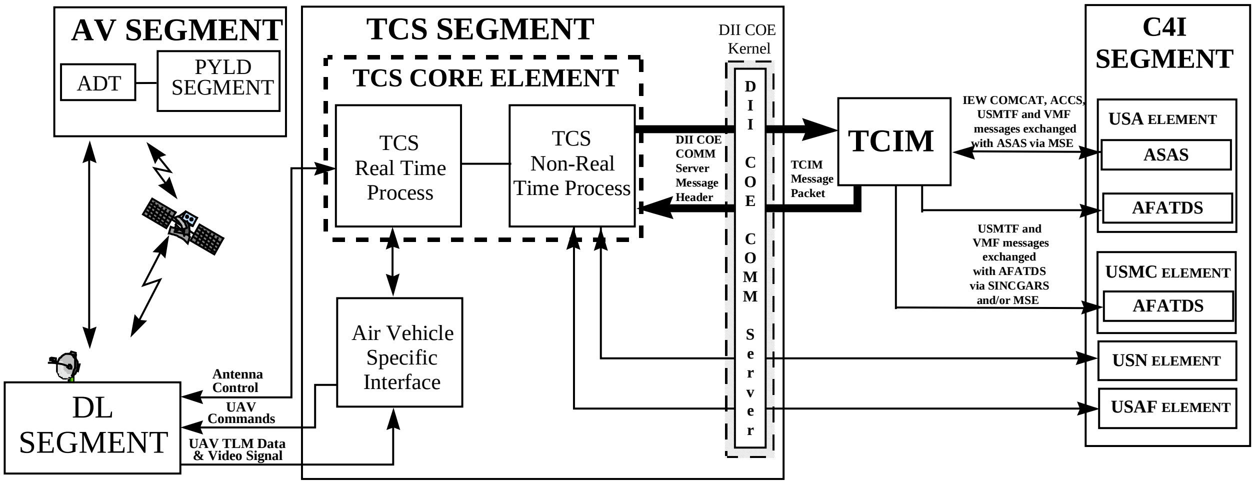
None.

3. Interface Design

The TCIM will provide the capability for TCS to exchange IEWCOMCAT, ACCS, USMTF, and VMF messages with ASAS and AFATDS via MSE and/or SINCGARS. The ASAS and AFATDS can communicate with TCS through MSE, using four-wire wireline and either the Digital Non-secure Voice Terminal (DNVT) or the Digital Subscriber Voice Terminal (DSVT) equipment suite. The AFATDS may also communicate with TCS through SINCGARS.

3.1 Interface Identification/Diagram

Figure 3.1-1 shows where TCIM fits into the TCS standard block diagram.


AV SEGMENT
ADT
PYLD SEGMENT
TCS SEGMENT
TCS CORE ELEMENT
TCS Real Time Process
TCS Non-Real Time Process
Air Vehicle Specific Interface
DII COE Kernel
DII COE COMM Server
DII COE COMM Server Message Header
TCIM Message Packet
TCIM
IEW COMCAT, ACCS, USMTF and VMF messages exchanged with ASAS via MSE
USMTF and VMF messages exchanged with AFATDS via SINCGARS and/or MSE
C4I SEGMENT
USA ELEMENT
ASAS
AFATDS
USMC ELEMENT
AFATDS
USN ELEMENT
USAF ELEMENT
DL SEGMENT
Antenna Control
UAV Commands
UAV TLM Data & Video Signal

BOLD INTERFACE LINE IDENTIFIES INTERFACE DEFINED IN THIS DOCUMENT

Figure 3.1-1 TCS Standard Block Interface Diagram with TCIM

3.2 TCS to TCIM Interface

The TCS software will use the DII COE Comm Server to communicate with the TCIM, hence this discussion focuses on the Application Program Interface (API) interface between TCS and the DII COE Comm Server.

The Comm Server provides automatic protocol conversion from a generic message format to any TCIM-supported protocol, and sends converted messages to the TCIM Common Interface Software (TCIS) for communication through the TCIMs. The Comm Server also enables host computers (Clients) without TCIMs to send/receive messages through TCIMs attached to other host (Server) platforms in the DII COE CS network.

The DII COE Comm Server will use the **tcomm_read()** and **tcomm_write()** library function calls as the API to communicate with TCS. The **tcomm_read()** and **tcomm_write()** functions use shared memory and Unix message queues to pass up to 1.5 megabytes of data between the

Comm Server and TCS. An operational overview of the Comm Server, including descriptions of the **tcomm_read()** and **tcomm_write()** functions is contained in Appendix A.

The Comm Server uses message headers to recognize the type of message to process. The following message headers have been defined:

- The Generic Message Header - Data Request
- The Generic Message Header - Data Indication
- The Generic Message Header - Data Status
- The Generic Message Header - TCIS Message
- The Variable Format Message Header - Data Request
- The Variable Format Message Header - Data Indication
- The Variable Format Message Header - Data Status

For the convenience of the reader, the information in the tables in the following paragraphs has been repeated from the *Interface Design Document (IDD) for the Communications Server Computer Software Component (CSC) of the Common Operating Environment (COE) Communications Software (CS),* written by Litton Data Systems, San Diego, CA 92121. The tables explain the structure of the headers. Note that the actual message data is at the end of each header.

3.2.1 Priority of Communications Interface

The TCIM has two distinct channels that are independently configured. The DII COE Comm Server establishes a unique message queue for each of the TCIM channels s to handle information exchange. Since each channel operates independently along a unique communication path, and the content of the TCIM message won't be analyzed until the message is received by the TCS application, there is no need to establish a protocol priority.

3.2.1.1 TCS Priority

(Not applicable to these interfaces.)

3.2.1.2 Data Link Priority

(Not applicable to these interfaces.)

3.2.2 Type of Communications Interface

The TCS host computer and the external TCIM will be interconnected by a Small Computer System Interface (SCSI) cable attached to the channel-1 SCSI port on the TCIM with a SCSI type 1 connector and a SCSI port on the TCS host computer. If the TCIM is the last device in the SCSI chain, it must have a SCSI terminator.

There are three types of software interfaces to be considered: TCIM initialization, message exchange with DII COE compliant systems, and message exchange with non-DII COE compliant legacy systems.

3.2.2.1 TCIM Initialization

The DII COE Comm Server supports a TCIM initialization tool called TCIM Configuration and Diagnostics Application (TC&D). TC&D provides full manual access to the capabilities of the TCIM through X Windows on POSIX systems. The user activates TC&D which leads the user through a series of dialogs in which the user selects the TCIM configuration options (e.g. protocol, bandwidth, etc.) to be used to initialize the TCIM. TC&D passes the TCIM initialization message to the DII COE Comm Server through the Comm Server API. The Comm Server initializes the TCIM and returns a TCIM status message to TC&D.

TC&D is provided with the TCIM or is available separately from the Army Common Hardware Software (CHS) Program Management Office (PMO). TC&D is an approved and tested Government Off-The-Shelf (GOTS) application.

3.2.2.2 Communication with DII COE Compliant Systems

The Comm Server will use the following message headers to exchange information with DII COE compliant systems

- The Generic Message Header - Data Request
- The Generic Message Header - Data Indication
- The Generic Message Header - Data Status
- The Variable Format Message Header - Data Request
- The Variable Format Message Header - Data Indication
- The Variable Format Message Header - Data Status

Figure 3.2.2.2-1 shows an example data flow for communication between compliant applications using SINCGARS.

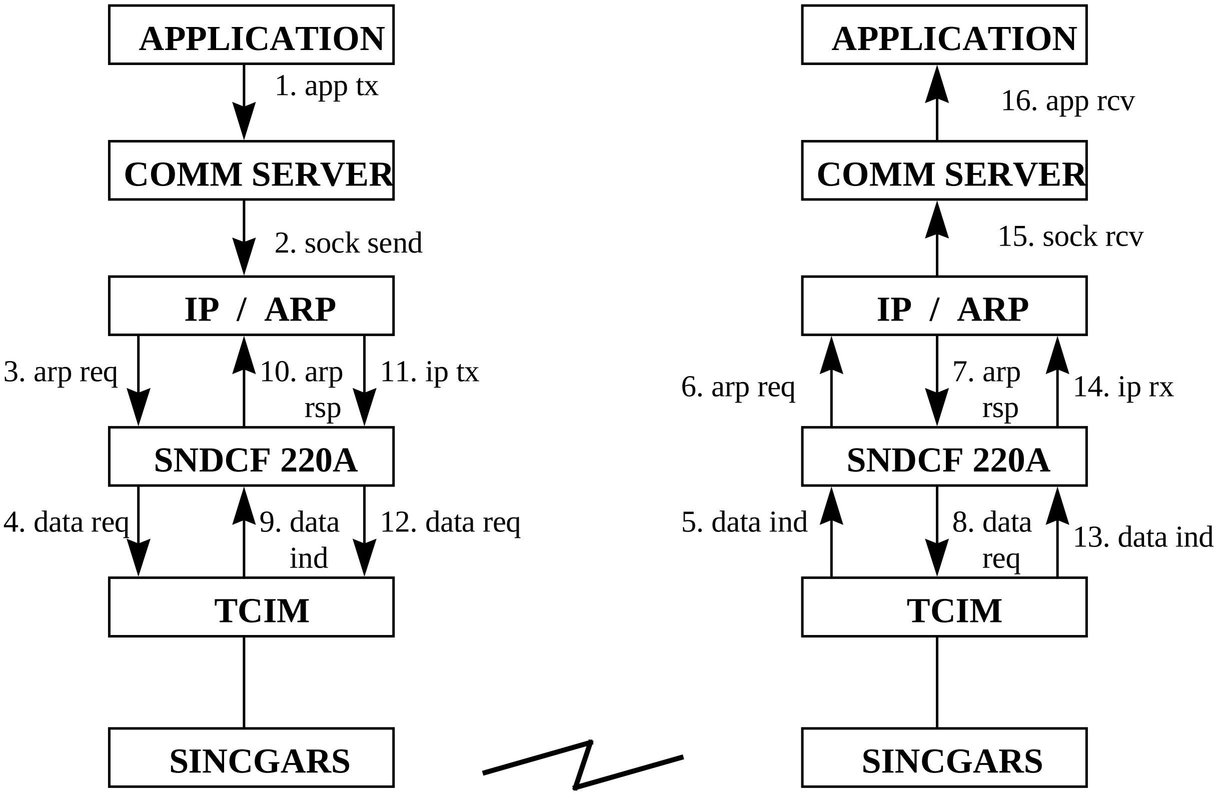


Figure 3.2.2.2-1. Data flow between TCS and DII COE Compliant Applications

3.2.2.2.1 Generic Message Header - Data Request

The Generic Message Header - Data Request is used by TCS to send messages to a remote host. The message is sent from TCS to the Comm Server. Table 3.2.2.2.1-1 shows the format for this message. Note: any value prefixed with “0x” (e.g. 0x2A) is a hexadecimal value.

Table 3.2.2.2.1-1 Generic Message Header - Data Request

Byte	Field Definition	Field Values	Comment
1	Message Type	0x01	
2	UDF (MSB)	binary	
3	UDF (LSB)	binary	
4	Reserved		
5 to 68	Source ID	ASCII Chars	64 bytes max
69	S_Appl_Q	84-xx	Q key of source app
70	# of Destinations	n=1-42	
71 to 134	Destination ID	ASCII chars	64 bytes max
135	D_Appl_Q	84-xx	Q key of destination app
136	Precedence	1-255	
137	Security	1-255	
138 to 143	Msg ID	6 bytes ASCII	

Byte	Field Definition	Field Values	Comment
144 to 148	Msg S/N	5 bytes ASCII	
149 to 160	DTG	12 bytes ASCII	
161	Connection Type		
[((n-1)*91)+162] to [((n-1)*91)+165]	Message Length	max = 1.5Mb	
[((n-1)*91)+166] to [((n-1)*91) + MsgLen -1]	Message Data		

Note: repeat bytes 71-161 for each destination

The valid TCIM Connection Types (byte 161) are listed in table 3.2.2.2.1-2

Table 3.2.2.2.1-2 TCIM Connection Types

Connection Type	Definition
100	**MTS**
101	**MTS-5**
102	**IFSS**
103	**ADDS**
104	**NRPT**
105	**NRPTAWE**
106	**GDU**
107	**AFCS**
255	**Automatic**

3.2.2.2.2 Generic Message Header - Data Indication

The Generic Message Header - Data Indication is used by the Comm Server to send incoming messages to TCS. Table 3.2.2.2.2-1 shows the format for this message. Note: any value prefixed with “0x” (e.g. 0x2A) is a hexadecimal value.

Table 3.2.2.2.2-1 Generic Message Header - Data Indication

Byte	Field Definition	Field Values	Comment
1	Message Type	0x02	
2	UDF (MSB)	binary	
3	UDF (LSB)	binary	
4	Reserved		
5 to 68	Source ID	ASCII Chars	64 bytes max
69	S_Appl_Q	84-xx	Q key of source app
70 to 133	Destination ID	ASCII chars	64 bytes max
134	D_Appl_Q	84-xx	Q key of dest. app
135	Precedence	1-255	
136	Security	1-255	

Byte	Field Definition	Field Values	Comment
137 to 142	Msg ID	6 bytes ASCII	
143 to 147	Msg S/N	5 bytes ASCII	
148 to 159	DTG	12 bytes ASCII	
160	Connection Type		
161 to 164	Message Length		
165 to n	Message Data		

3.2.2.2.3 Generic Message Header - Data Status

The Generic Message Header - Data Status message is used by the Comm Server to inform an application of the communications status of a message (e.g., ACK, NAK, no response). The message is sent from the Comm Server to TCS. Table 3.2.2.2.3-1 shows the format for this message. Table 3.2.2.2.3-2 shows the valid status codes to be placed into byte 158 for this message. Note: any value prefixed with “0x” (e.g. 0x2A) is a hexadecimal value.

Table 3.2.2.2.3-1 Generic Message Header - Data Status

Byte	Field Definition	Field Values	Comment
1	Message Type	0x03	
2	UDF (MSB)	binary	
3	UDF (LSB)	binary	
4	Reserved		
5 to 68	Source ID	ASCII Chars	64 bytes max
69	S_Appl_Q	84-xx	Q key of source app
70 to 133	Destination ID	ASCII chars	64 bytes max
134	D_Appl_Q	84-xx	Q key of destination app
135 to 140	Msg ID	6 bytes ASCII	
141 to 145	Msg S/N	5 bytes ASCII	
146 to 157	DTG	12 bytes ASCII	
158	Status		

Table 3.2.2.2.3-2 Status Codes

Status Code	Definition
0x00	ACK
0x01	NAK
0x02	No Response
0x03	Failed Transmission - Subscribers Failure
0x04	Failed Transmission - No link established
0x05	Failed Transmission - Bad Configuration
0x07	Failed Transmission - Missing clock (KY-57, external)
0x08	Failed Transmission - Time expired
0x09	Failed Transmission - Radio silence ON
0x0B	Failed Transmission - Retry failure

Status Code	Definition
0x0C	Failed Transmission - Link busy
0x0D	Message queued
0x10	Format error
0x11	Wrong State
0x12	Invalid Protocol
0x13	SCSI I/O Control error
0x14	SCSI Transmit error
0x15	TCIM Buffer Full
0x16	Bus receive error
0x17	System error
0x18	Device failed
0x1A	Unsupported values
0x1B	Out of Memory
0x1C	Invalid Source ID
0x1D	Invalid Destination ID
0x1E	Invalid Logical Unit
0x20	Receiver ready
0x21	Receiver not ready
0x22	Transmission complete
0x23	No resources
0x24	Invalid MAIL name
0x25	No Sendmail on System

3.2.2.2.4 Variable Message Format Header - Data Request

The Variable Message Format Header - Data Request message is used by TCS to send VMF messages to a remote host. This message is sent from TCS to the Comm Server. Table 3.2.2.2.4-1 shows the format for this message. Note: any value prefixed with “0x” (e.g. 0x2A) is a hexadecimal value.

Table 3.2.2.2.4-1 Variable Message Format Header - Data Request

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
2	Header Version	0=MIL-STD-2045-47001 1=MIL-STD-2045-47001B	binary (0-16)
3 to 4	Status Tracking	1 to 65535	MSB-LSB
5	Connection type	Binary (1 to 255)	Refer to table 3.2.2.2.1-2
6	Data Compression Indicator	0=No Compression 1=Comm Server compression 2=Unix Compression	binary
7 to 8	Reserved	constant = 0	for word alignment

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
Originator Address Group (G1) Note: The following Fields in G1 will be appended after the Reserved field above.			
1	G1 Address Present Indicator	0= Address not present 1=Address present	binary
2 to 4	Reserved	constant = 0	for word alignment
If G1 Address Present Indicator = 0			
5 to 72	Reserved	constant = 0	for word alignment
If G1 Address Present Indicator = 1			
5	G1 URN indicator	0=Unit name is used 1=Unit Ref# is used	binary
6 to 8	Reserved	constant = 0	for word alignment
If G1 URN Indicator = 0			
9 to 72	Unit Name	ASCII	Max 64 characters
If G1 URN Indicator = 1			
9 to 12	Unit Ref#	0x00 to 0xFFFFFF	binary (MSB-LSB)
13 to 72	Reserved	constant = 0	for word alignment
Recipient Address Group (R1) Note: The following Fields in R1 will be appended after the last field of the G1 group.			
1	Total Recipient #	0=No recipient 1 to 16=# of recipients	binary
2 to 4	Reserved	constant = 0	for word alignment
The following recipient entry bytes [flagged by *] are repeated for R = 0 through 15			
* 5+(Rx68)	R1 Urn indicator	0=Unit name is used 1=Unit Ref# is used	binary
* 6+(Rx68) to 8+(Rx68)	Reserved	constant = 0	for word alignment
If R1 URN Indicator = 0			
* 9+(Rx68) to 72+(Rx68)	Unit Name	ASCII	Max 64 characters
If R1 URN Indicator = 1			
* 9+(Rx68) to 12+(Rx68)	Unit Ref#	0x00 to 0xFFFFFF	binary (MSB-LSB)
* 13+(Rx68) to 72+(Rx68)	Reserved	constant = 0	for word alignment
Information Address Group (R2) Note: The following Fields in R2 will be appended after the last field of the R1 group.			
1	Total Information Recipient #	0=No recipient 1 to 16=# of recipients	binary
2 to 4	Reserved	constant = 0	for word alignment
The following recipient entry bytes [flagged by *] are repeated for I = 0 through 15			
* 5+(Ix68)	R2 Urn indicator	0=Unit name is used 1=Unit Ref# is used	binary
* 6+(Ix68) to 8+(Ix68)	Reserved	constant = 0	for word alignment
If R2 URN Indicator = 0			
* 9+(Ix68) to 72+(Ix68)	Unit Name	ASCII	Max 64 characters
If R2 URN Indicator = 1			
* 9+(Ix68) to 12+(Ix68)	Unit Ref#	0x00 to 0xFFFFFF	binary (MSB-LSB)
* 13+(Ix68) to 72+(Ix68)	Reserved	constant = 0	for word alignment

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
Message Handler Group (R3) Note: The following Fields in R3 will be appended after the last field of the R2 group.			
1	Total Concat Message #	0=NoConcat messages 1 to 16=# of Concat messages	binary
2 to 4	Reserved	constant = 0	for word alignment
The following Concatenated message entry bytes [flagged by *] are repeated for C = 0 through 15			
* 5+(Cx7464) to 8+(Cx7464)	Message Size	D = 0 to 1megabyte	binary (MSB-LSB)
* 9+(Cx7464)	User Message Format	0 = TADILJ 1= Binary File 2 = VMF 3 = NITFS 4 = Forwarded Message 5 = USMTF 6 = DOI 103	binary
* 10 + (Cx7464)	R3 Functional Area Designator	0 to 15	binary
* 11 + (Cx7464)	R3 Message Number	1 to 127	binary
* 12 + (Cx7464)	Operation Indicator	0 = Operation 1 = Exercise 2 = Simulation 3 = Undefined Op	binary
* 13 + (Cx7464)	Retransmit Indicator	0 = Not Retransmit 1 = Retransmit	binary
* 14 + (Cx7464)	Message Precedence	2 = Emergency Command 4 = Flash 5 = Immediate 6 = Priority 7 = Routine	binary
* 15 + (Cx7464)	Message Security	0 = Unclassified 1 = Confidential 2 = Secret 3 = Top Secret	binary
* 16 + (Cx7464)	Message Option ID	0 = No optional fields 1 = Message subtype# 2 = Control/Release 3 = Message subtype# and Control/ Release 4 = File Name 5 = Message subtype# and File Name 7 = All optional fields are used	binary
* 17 + (Cx7464)	Message Subtype #	0 to 127	binary
* 18 + (Cx7464) to 19 + (Cx7464)	Control/Release Marking	ASCII	2 Characters max

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
* 20 + (Cx7464)	Reserved	constant = 0	for word alignment
* 21 + (Cx7464) to 84 + (Cx7464)	File Name	ASCII	64 characters max
* 85 + (Cx7464)	Total # of R3 additional fields	R3A = 0 to 16	binary
* 86 + (Cx7464) to 88 + (Cx7464)	Reserved	constant = 0	for word alignment
The following additional fields entry bytes [flagged by !] are repeated for R3A = 0 through 15 and will be appended after the reserved field above			
! 1+(R3Ax72)	Bit Byte Size	0 = Bit Field 1 = Character Field	binary
! 2+(R3Ax72)	Field Size	0 to 255 (bits/char)	binary
! 3+(R3Ax72) to 4+(R3Ax72)	Reserved	constant = 0	for word alignment
! 5+(R3Ax72)	R3 Codeword Indicator	0 = Bit codeword 1 = Character codeword	binary
! 6+(R3Ax72) to 8+(R3Ax72)	Reserved	constant = 0	for word alignment
If R3 Codeword Indicator = 0			
! 9+(R3Ax72) to 12+(R3Ax72)	Bit Codeword	32 bit number	binary (MSB-LSB)
! 13+(R3Ax72) to 72+(R3Ax72)	Reserved	constant = 0	for word alignment
If R3 Codeword Indicator = 1			
! 9+(R3Ax72) to 72+(R3Ax72)	Character Codeword	ASCII	Max 64 characters
Appended to R3 Codeword			

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
* 1240 + (Cx7464)	R3 Present Group Indicator	1 = Orig DTG 2 = Perish DTG 3 = Orig DTG and Perish DTG 4 = ACK group 5 = Orig DTG and ACK Group 7 = Orig DTG and Perish DTG and ACK group 8 = Resp Data Grp 9 = Orig DTG and Resp Data 16 = Refer Data Grp 17 = Orig DTG and Refer Grp 19 = Orig DTG and Perish DTG and Refer Grp 32 = Msg Ident Grp 33 = Orig DTG and Msg Ident Grp 34 Perish DTG and Msg Ident Grp 35 = Orig DTG and Perish DTG and Msg Ident Grp 36 = ACK Grp and Msg Ident Grp 37 = Orig DTG and ACK Grp and Msg Ident Grp 39 = Orig DTG and Perish DTG and ACK Grp and Msg Ident Grp 40 = Resp Data Grp and Msg Ident 41 = Orig DTG and Resp Data Grp and Msg Ident 48 = Refer Data Grp and Msg Ident 49 = Orig DTG and Refer Data Grp and Msg Ident 51 = Orig DTG and Perish DTG and Refer Data Grp and Msg Ident	binary (The values for Orig DTG and Perish DTG and ACK Group and Resp Data Grp and Refer Data Grp and Msg Ident Grp can be OR’ed together
ACK Request Group (G7)			

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
* 1241 + (Cx7464)	ACK Req Indicator	1 = Machine ACK required 2 = Operator ACK required 4 = Operator Reply required	binary
* 1242 + (Cx7464) to 1244 + (Cx7464)	Reserved	constant = 0	for word alignment
Originator DTG Group (G5)			
* 1245 + (Cx7464)	G5 Year	0 to 99	binary
* 1246 + (Cx7464)	G5 Month	1 to 12	binary
* 1247 + (Cx7464)	G5 Day	1 to 31	binary
* 1248 + (Cx7464)	G5 Hour	0 to 23	binary
* 1249 + (Cx7464)	G5 Minute	0 to 59	binary
* 1250 + (Cx7464)	G5 Second	0 to 59	binary
* 1251 + (Cx7464)	Reserved	constant = 0	for word alignment
* 1252 + (Cx7464)	Extension Indicator	1 = extension used 0 = extension not used	binary
* 1253 + (Cx7464) to 1254 + (Cx7464)	extension	0x000 to 0xFFF (0 to 4095)	binary (MSB-LSB)
* 1255 + (Cx7464) to 1256 + (Cx7464)	Reserved	constant = 0	for word alignment
Perishability DTG Group (G6)			
* 1257 + (Cx7464)	G6 Year	0 to 99	binary
* 1258 + (Cx7464)	G6 Month	1 to 12	binary
* 1259 + (Cx7464)	G6 Day	1 to 31	binary
* 1260 + (Cx7464)	G6 Hour	0 to 23	binary
* 1261 + (Cx7464)	G6 Minute	0 to 59	binary
* 1262 + (Cx7464)	G6 Second	0 to 59	binary
* 1263 + (Cx7464) to 1268 + (Cx7464)	Reserved	constant = 0	for word alignment
Response Data Group (G8)			
* 1269 + (Cx7464)	G8 Year	0 to 99	binary
* 1270 + (Cx7464)	G8 Month	1 to 12	binary
* 1271 + (Cx7464)	G8 Day	1 to 31	binary
* 1272 + (Cx7464)	G8 Hour	0 to 23	binary
* 1273 + (Cx7464)	G8 Minute	0 to 59	binary
* 1274 + (Cx7464)	G8 Second	0 to 59	binary
* 1275 + (Cx7464)	Reserved	constant = 0	for word alignment
* 1276 + (Cx7464)	Extension Indicator	1 = extension used 0 = extension not used	binary
* 1277 + (Cx7464) to 1278 + (Cx7464)	extension	0x000 to 0xFFF (0 to 4095)	binary (MSB-LSB)
* 1279 + (Cx7464) to 1280 + (Cx7464)	Reserved	constant = 0	for word alignment

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
* 1281 + (Cx7464)	Receipt/Compliance	1 = Machine Receipt 2 = Cannot process 3 = Operator ACK 4 = Will Comply 5 = have completed 6 = Cannot Comply	binary
* 1282 + (Cx7464)	Reply Amp ID	1 = CANTCO reason code 2 = CANTPRO reason code 4 = Reply Amp present 5 = CANTCO and Reply Amp present	binary
* 1283 + (Cx7464)	CANTCO Reason Code	0 = Comms problem 1 = Ammunition problem 2 = Personnel problem 3 = Fuel problem 4 = Terrain Env. Problem 5 = Equipment problem 6 = Tactical Situation problem 7 = Other reason	binary
* 1284 + (Cx7464)	CANTPRO Reason code	1 = CANTPRO code 1 2 = CANTPRO code 2 3 = CANTPRO code 3 4 = CANTPRO code 4 5 = CANTPRO code 5 6 = CANTPRO code 6 7 = CANTPRO code 7 8 = CANTPRO code 8 9 = CANTPRO code 9 10 = CANTPRO code 10 11 = CANTPRO code 11 12 = CANTPRO code 12 13 = CANTPRO code 13 14 = CANTPRO code 14 15 = CANTPRO code 15 16 = CANTPRO code 16 17 = CANTPRO code 17 18 = CANTPRO code 18 19 = CANTPRO code 19 20 = CANTPRO code 20 21 = CANTPRO code 21 22 = CANTPRO code 22 23 = CANTPRO code 23 24 = CANTPRO code 24 25 = CANTPRO code 25	binary

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
* 1285 + (Cx7464) to 1334 + (Cx7464)	Reply Amplification	ASCII	50 chars max
* 1335 + (Cx7464) to 1336 + (Cx7464)	Reserved	constant = 0	for word alignment
Reference Message Data Group (G9)			
* 1337 + (Cx7464)	Total # of Refer Data	R4 = 0 to 4	binary
* 1338 + (Cx7464) to 1340 + (Cx7464)	Reserved	constant = 0	for word alignment
The following reference message entry bytes [flagged by !] are repeated for R4 = 0 through 3 and will be appended after the reserved field above			
! 1 + (R4x1243)	G9 Year	0 to 99	binary
! 2 + (R4x1243)	G9 Month	1 to 12	binary
! 3 + (R4x1243)	G9 Day	1 to 31	binary
! 4 + (R4x1243)	G9 Hour	0 to 23	binary
! 5 + (R4x1243)	G9 Minute	0 to 59	binary
! 6 + (R4x1243)	G9 Second	0 to 59	binary
! 7 + (R4x1243)	Reserved	constant = 0	for word alignment
! 8 + (R4x1243)	Extension Indicator	1 = extension used 0 = extension not used	binary
! 9 + (R4x1243) to 10 + (R4x1243)	extension	0x000 to 0xFFF (0 to 4095)	binary (MSB-LSB)
! 11 + (R4x1243) to 12 + (R4x1243)	Reserved	constant = 0	for word alignment
! 13 + (R4x1243)	G9 Function area	0 to 15	binary
! 14 + (R4x1243)	G9 Message Number	1 to 127	binary
! 15 + (R4x1243) to 16 + (R4x1243)	Reserved	constant = 0	for word alignment
! 17 + (R4x1243)	G9 Address Present Indicator	0 = Address not present 1 = Address present	binary
! 18 + (R4x1243) to 20 + (R4x1243)	Reserved	constant = 0	for word alignment
If G9 Address present Indicator = 1			
! 21 + (R4x1243)	G9 URN Indicator	0 = Unit Name is used 1 = Unit Ref# is used	binary
! 22 + (R4x1243) to 23 + (R4x1243)	Reserved	constant = 0	for word alignment
If G9 URN Indicator = 0			
! 24 + (R4x1243) to 87 + (R4x1243)	Unit Name	ASCII	64 Chars max
If G9 URN Indicator = 1			
! 24 + (R4x1243) to 27 + (R4x1243)	Unit Reference #	0x00 to 0xFFFFFF	binary (MSB-LSB)
! 28 + (R4x1243) to 87 + (R4x1243)	Reserved	constant = 0	for word alignment
! 88 + (R4x1243)	Total of R4 Additional fields	R4A = 1 to 16	binary
! 89 + (R4x1243) to 91 + (R4x1243)	Reserved	constant = 0	for word alignment
The following reference message entry bytes [flagged by @] are repeated for R4A = 0 through 15 and will be appended after the reserved field above			

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
@ 1+(R4Ax72)	Bit Byte Size	0 = Bit Field 1 = Character Field	binary
@ 2+(R4Ax72)	Field Size	0 to 255 (bits/char)	binary
@ 3+(R4Ax72) to 4+(R4Ax72)	Reserved	constant = 0	for word alignment
@ 5+(R4Ax72)	R4 Codeword Indicator	0 = Bit codeword 1 = Character codeword	binary
@ 6+(R4Ax72) to 8+(R4Ax72)	Reserved	constant = 0	for word alignment
If R4 Codeword Indicator = 0			
@ 9+(R4Ax72) to 12+(R4Ax72)	Bit Codeword	16 bit number	binary (MSB-LSB)
@ 13+(R4Ax72) to 72+(R4Ax72)	Reserved	constant = 0	for word alignment
If R4 Codeword Indicator = 1			
@ 9+(R4Ax72) to 72+(R4Ax72)	Character Codeword	ASCII	Max 64 characters
Appended to R4 Codeword			
0 to 1 megabyte	Data	T = 0 to 1 Megabyte	Binary

3.2.2.2.5 Variable Message Format Header - Data Indication

The Variable Message Format Header - Data indication message is used by the Comm Server to send an incoming VMF message to TCS. Table 3.2.2.2.5-1 shows the format for this message. Note: any value prefixed with "0x" (e.g. 0x2A) is a hexadecimal value.

Table 3.2.2.2.5-1 Variable Message Format Header - Data Indication

Byte	Field Definition	Field Values	Comment
1	Message Type	0x22	
2	Header Version	0=MIL-STD-2045-47001 1=MIL-STD-2045-47001B	binary (0-16)
3 to 4	Status Tracking	1 to 65535	MSB-LSB
5	Connection type	Binary (1 to 255)	Refer to table 3.2.2.2.1-2
6	Data Compression Indicator	0=No Compression 1=Comm Server compression 2=Unix Compression	binary
7 to 8	Reserved	constant = 0	for word alignment
Originator Address Group (G1) Note: The following Fields in G1 will be appended after the Reserved field above.			
1	G1 Address Present Indicator	0= Address not present 1=Address present	binary
2 to 4	Reserved	constant = 0	for word alignment
If G1 Address Present Indicator = 0			
5 to 72	Reserved	constant = 0	for word alignment
If G1 Address Present Indicator = 1			
5	G1 URN indicator	0=Unit name is used 1=Unit Ref# is used	binary
6 to 8	Reserved	constant = 0	for word alignment
If G1 URN Indicator = 0			
9 to 72	Unit Name	ASCII	Max 64 characters
If G1 URN Indicator = 1			
9 to 12	Unit Ref#	0x00 to 0xFFFFFF	binary (MSB-LSB)
13 to 72	Reserved	constant = 0	for word alignment
Recipient Address Group (R1) Note: The following Fields in R1 will be appended after the last field of the G1 group.			
1	Total Recipient #	0=No recipient 1 to 16=# of recipients	binary
2 to 4	Reserved	constant = 0	for word alignment
The following recipient entry bytes [flagged by *] are repeated for R = 0 through 15			
* 5+(Rx68)	R1 URN indicator	0=Unit name is used 1=Unit Ref# is used	binary
* 6+(Rx68) to 8+(Rx68)	Reserved	constant = 0	for word alignment
If R1 URN Indicator = 0			
* 9+(Rx68) to 72+(Rx68)	Unit Name	ASCII	Max 64 characters
If R1 URN Indicator = 1			

Byte	Field Definition	Field Values	Comment
* 9+(Rx68) to 12+(Rx68)	Unit Ref#	0x00 to 0xFFFFFF	binary (MSB-LSB)
* 13+(Rx68) to 72+(Rx68)	Reserved	constant = 0	for word alignment
Information Address Group (R2) Note: The following Fields in R2 will be appended after the last field of the R1 group.			
1	Total Information Recipient #	0=No recipient 1 to 16=# of recipients	binary
2 to 4	Reserved	constant = 0	for word alignment
The following recipient entry bytes [flagged by *] are repeated for I = 0 through 15			
* 5+(Ix68)	R2 Urn indicator	0=Unit name is used 1=Unit Ref# is used	binary
* 6+(Ix68) to 8+(Ix68)	Reserved	constant = 0	for word alignment
If R2 URN Indicator = 0			
* 9+(Ix68) to 72+(Ix68)	Unit Name	ASCII	Max 64 characters
If R2 URN Indicator = 1			
* 9+(Ix68) to 12+(Ix68)	Unit Ref#	0x00 to 0xFFFFFF	binary (MSB-LSB)
* 13+(Ix68) to 72+(Ix68)	Reserved	constant = 0	for word alignment
Message Handler Group (R3) Note: The following Fields in R3 will be appended after the last field of the R2 group.			
1	Total Concat Message #	0=NoConcat messages 1 to 16=# of Concat messages	binary
2 to 4	Reserved	constant = 0	for word alignment
The following Concatenated message entry bytes [flagged by *] are repeated for C = 0 through 15			
* 5+(Cx7464) to 8+(Cx7464)	Message Size	D = 0 to 1megabyte	binary (MSB-LSB)
* 9+(Cx7464)	User Message Format	0 = TADILJ 1= Binary File 2 = VMF 3 = NITFS 4 = Forwarded Message 5 = USMTF 6 = DOI 103	binary
* 10 + (Cx7464)	R3 Functional Area Designator	0 to 15	binary
* 11 + (Cx7464)	R3 Message Number	1 to 127	binary
* 12 + (Cx7464)	Operation Indicator	0 = Operation 1 = Exercise 2 = Simulation 3 = Undefined Op	binary
* 13 + (Cx7464)	Retransmit Indicator	0 = Not Retransmit 1 = Retransmit	binary
* 14 + (Cx7464)	Message Precedence	2 = Emergency Command 4 = Flash 5 = Immediate 6 = Priority 7 = Routine	binary

Byte	Field Definition	Field Values	Comment
* 15 + (Cx7464)	Message Security	0 = Unclassified 1 = Confidential 2 = Secret 3 = Top Secret	binary
* 16 + (Cx7464)	Message Option ID	0 = No optional fields 1 = Message subtype# 2 = Control/Release 3 = Message subtype# and Control/Release 4 = File Name 5 = Message subtype# and File Name 7 = All optional fields are used	binary
* 17 + (Cx7464)	Message Subtype #	0 to 127	binary
* 18 + (Cx7464) to 19 + (Cx7464)	Control/Release Marking	ASCII	2 Characters max
* 20 + (Cx7464)	Reserved	constant = 0	for word alignment
* 21 + (Cx7464) to 84 + (Cx7464)	File Name	ASCII	64 characters max
* 85 + (Cx7464)	Total # of R3 additional fields	R3A = 0 to 16	binary
* 86 + (Cx7464) to 88 + (Cx7464)	Reserved	constant = 0	for word alignment
The following additional fields entry bytes [flagged by !] are repeated for R3A = 0 through 15 and will be appended after the reserved field above			
! 1+(R3Ax72)	Bit Byte Size	0 = Bit Field 1 = Character Field	binary
! 2+(R3Ax72)	Field Size	0 to 255 (bits/char)	binary
! 3+(R3Ax72) to 4+(R3Ax72)	Reserved	constant = 0	for word alignment
! 5+(R3Ax72)	R3 Codeword Indicator	0 = Bit codeword 1 = Character codeword	binary
! 6+(R3Ax72) to 8+(R3Ax72)	Reserved	constant = 0	for word alignment
If R3 Codeword Indicator = 0			
! 9+(R3Ax72) to 12+(R3Ax72)	Bit Codeword	32 bit number	binary (MSB-LSB)
! 13+(R3Ax72) to 72+(R3Ax72)	Reserved	constant = 0	for word alignment
If R3 Codeword Indicator = 1			
! 9+(R3Ax72) to 72+(R3Ax72)	Character Codeword	ASCII	Max 64 characters
Appended to R3 Codeword			

Byte	Field Definition	Field Values	Comment
* 1240 + (Cx7464)	R3 Present Group Indicator	1 = Orig DTG 2 = Perish DTG 3 = Orig DTG and Perish DTG 4 = ACK group 5 = Orig DTG and ACK Group 7 = Orig DTG and Perish DTG and ACK group 8 = Resp Data Grp 9 = Orig DTG and Resp Data 16 = Refer Data Grp 17 = Orig DTG and Refer Grp 19 = Orig DTG and Perish DTG and Refer Grp 32 = Msg Ident Grp 33 = Orig DTG and Msg Ident Grp 34 Perish DTG and Msg Ident Grp 35 = Orig DTG and Perish DTG and Msg Ident Grp 36 = ACK Grp and Msg Ident Grp 37 = Orig DTG and ACK Grp and Msg Ident Grp 39 = Orig DTG and Perish DTG and ACK Grp and Msg Ident Grp 40 = Resp Data Grp and Msg Ident 41 = Orig DTG and Resp Data Grp and Msg Ident 48 = Refer Data Grp and Msg Ident 49 = Orig DTG and Refer Data Grp and Msg Ident 51 = Orig DTG and Perish DTG and Refer Data Grp and Msg Ident	binary (The values for Orig DTG and Perish DTG and ACK Group and Resp Data Grp and Refer Data Grp and Msg Ident Grp can be OR'ed together
ACK Request Group (G7)			

Byte	Field Definition	Field Values	Comment
* 1241 + (Cx7464)	ACK Req Indicator	1 = Machine ACK required 2 = Operator ACK required 4 = Operator Reply required	binary
* 1242 + (Cx7464) to 1244 + (Cx7464)	Reserved	constant = 0	for word alignment
Originator DTG Group (G5)			
* 1245 + (Cx7464)	G5 Year	0 to 99	binary
* 1246 + (Cx7464)	G5 Month	1 to 12	binary
* 1247 + (Cx7464)	G5 Day	1 to 31	binary
* 1248 + (Cx7464)	G5 Hour	0 to 23	binary
* 1249 + (Cx7464)	G5 Minute	0 to 59	binary
* 1250 + (Cx7464)	G5 Second	0 to 59	binary
* 1251 + (Cx7464)	Reserved	constant = 0	for word alignment
* 1252 + (Cx7464)	Extension Indicator	1 = extension used 0 = extension not used	binary
* 1253 + (Cx7464) to 1254 + (Cx7464)	extension	0x000 to 0xFFF (0 to 4095)	binary (MSB-LSB)
* 1255 + (Cx7464) to 1256 + (Cx7464)	Reserved	constant = 0	for word alignment
Perishability DTG Group (G6)			
* 1257 + (Cx7464)	G6 Year	0 to 99	binary
* 1258 + (Cx7464)	G6 Month	1 to 12	binary
* 1259 + (Cx7464)	G6 Day	1 to 31	binary
* 1260 + (Cx7464)	G6 Hour	0 to 23	binary
* 1261 + (Cx7464)	G6 Minute	0 to 59	binary
* 1262 + (Cx7464)	G6 Second	0 to 59	binary
* 1263 + (Cx7464) to 1268 + (Cx7464)	Reserved	constant = 0	for word alignment
Response Data Group (G8)			
* 1269 + (Cx7464)	G8 Year	0 to 99	binary
* 1270 + (Cx7464)	G8 Month	1 to 12	binary
* 1271 + (Cx7464)	G8 Day	1 to 31	binary
* 1272 + (Cx7464)	G8 Hour	0 to 23	binary
* 1273 + (Cx7464)	G8 Minute	0 to 59	binary
* 1274 + (Cx7464)	G8 Second	0 to 59	binary
* 1275 + (Cx7464)	Reserved	constant = 0	for word alignment
* 1276 + (Cx7464)	Extension Indicator	1 = extension used 0 = extension not used	binary
* 1277 + (Cx7464) to 1278 + (Cx7464)	extension	0x000 to 0xFFF (0 to 4095)	binary (MSB-LSB)
* 1279 + (Cx7464) to 1280 + (Cx7464)	Reserved	constant = 0	for word alignment
* 1281 + (Cx7464)	Receipt/Compliance	1 = Machine Receipt 2 = Cannot process 3 = Operator ACK 4 = Will Comply 5 = have completed 6 = Cannot Comply	binary

Byte	Field Definition	Field Values	Comment
* 1282 + (Cx7464)	Reply Amp ID	1 = CANTCO reason code 2 = CANTPRO reason code 4 = Reply Amp present 5 = CANTCO and Reply Amp present	binary
* 1283 + (Cx7464)	CANTCO Reason Code	0 = Comms problem 1 = Ammunition problem 2 = Personnel problem 3 = Fuel problem 4 = Terrain Env. Problem 5 = Equipment problem 6 = Tactical Situation problem 7 = Other reason	binary
* 1284 + (Cx7464)	CANTPRO Reason code	1 = CANTPRO code 1 2 = CANTPRO code 2 3 = CANTPRO code 3 4 = CANTPRO code 4 5 = CANTPRO code 5 6 = CANTPRO code 6 7 = CANTPRO code 7 8 = CANTPRO code 8 9 = CANTPRO code 9 10 = CANTPRO code 10 11 = CANTPRO code 11 12 = CANTPRO code 12 13 = CANTPRO code 13 14 = CANTPRO code 14 15 = CANTPRO code 15 16 = CANTPRO code 16 17 = CANTPRO code 17 18 = CANTPRO code 18 19 = CANTPRO code 19 20 = CANTPRO code 20 21 = CANTPRO code 21 22 = CANTPRO code 22 23 = CANTPRO code 23 24 = CANTPRO code 24 25 = CANTPRO code 25	binary
* 1285 + (Cx7464) to 1334 + (Cx7464)	Reply Amplification	ASCII	50 chars max
* 1335 + (Cx7464) to 1336 + (Cx7464)	Reserved	constant = 0	for word alignment
Reference Message Data Group (G9)			
* 1337 + (Cx7464)	Total # of Refer Data	R4 = 0 to 4	binary

Byte	Field Definition	Field Values	Comment
* 1338 + (Cx7464) to 1340 + (Cx7464)	Reserved	constant = 0	for word alignment
The following reference message entry bytes [flagged by !] are repeated for R4 = 0 through 3 and will be appended after the reserved field above			
! 1 + (R4x1243)	G9 Year	0 to 99	binary
! 2 + (R4x1243)	G9 Month	1 to 12	binary
! 3 + (R4x1243)	G9 Day	1 to 31	binary
! 4 + (R4x1243)	G9 Hour	0 to 23	binary
! 5 + (R4x1243)	G9 Minute	0 to 59	binary
! 6 + (R4x1243)	G9 Second	0 to 59	binary
! 7 + (R4x1243)	Reserved	constant = 0	for word alignment
! 8 + (R4x1243)	Extension Indicator	1 = extension used 0 = extension not used	binary
! 9 + (R4x1243) to 10 + (R4x1243)	extension	0x000 to 0xFFF (0 to 4095)	binary (MSB-LSB)
! 11 + (R4x1243) to 12 + (R4x1243)	Reserved	constant = 0	for word alignment
! 13 + (R4x1243)	G9 Function area	0 to 15	binary
! 14 + (R4x1243)	G9 Message Number	1 to 127	binary
! 15 + (R4x1243) to 16 + (R4x1243)	Reserved	constant = 0	for word alignment
! 17 + (R4x1243)	G9 Address Present Indicator	0 = Address not present 1 = Address present	binary
! 18 + (R4x1243) to 20 + (R4x1243)	Reserved	constant = 0	for word alignment
If G9 Address present Indicator = 1			
! 21 + (R4x1243)	G9 URN Indicator	0 = Unit Name is used 1 = Unit Ref# is used	binary
! 22 + (R4x1243) to 23 + (R4x1243)	Reserved	constant = 0	for word alignment
If G9 URN Indicator = 0			
! 24 + (R4x1243) to 87 + (R4x1243)	Unit Name	ASCII	64 Chars max
If G9 URN Indicator = 1			
! 24 + (R4x1243) to 27 + (R4x1243)	Unit Reference #	0x00 to 0xFFFFFF	binary (MSB-LSB)
! 28 + (R4x1243) to 87 + (R4x1243)	Reserved	constant = 0	for word alignment
! 88 + (R4x1243)	Total of R4 Additional fields	R4A = 1 to 16	binary
! 89 + (R4x1243) to 91 + (R4x1243)	Reserved	constant = 0	for word alignment
The following reference message entry bytes [flagged by @] are repeated for R4A = 0 through 15 and will be appended after the reserved field above			
@ 1+(R4Ax72)	Bit Byte Size	0 = Bit Field 1 = Character Field	binary
@ 2+(R4Ax72)	Field Size	0 to 255 (bits/char)	binary
@ 3+(R4Ax72) to 4+(R4Ax72)	Reserved	constant = 0	for word alignment

Byte	Field Definition	Field Values	Comment
@ 5+(R4Ax72)	R4 Codeword Indicator	0 = Bit codeword 1 = Character codeword	binary
@ 6+(R4Ax72) to 8+(R4Ax72)	Reserved	constant = 0	for word alignment
If R4 Codeword Indicator = 0			
@ 9+(R4Ax72) to 12+(R4Ax72)	Bit Codeword	16 bit number	binary (MSB-LSB)
@ 13+(R4Ax72) to 72+(R4Ax72)	Reserved	constant = 0	for word alignment
If R4 Codeword Indicator = 1			
@ 9+(R4Ax72) to 72+(R4Ax72)	Character Codeword	ASCII	Max 64 characters
Appended to R4 Codeword			
0 to 1 megabyte	Data	T = 0 to 1 Megabyte	Binary

3.2.2.2.6 Variable Message Format Header - Data Status

The Variable Message Format Header - Data Status message is used by the Comm Server to inform an application of the communications status of a message (e.g., ACK, NAK, no response). The message is sent from the Comm Server to TCS. Table 3.2.2.2.6-1 shows the format for this message. Table 3.2.2.2.6-2 shows the valid status codes for this message. Note: any value prefixed with “0x” (e.g. 0x2A) is a hexadecimal value.

Table 3.2.2.2.6-1 Variable Message Format Header - Data Status

Byte	Field Definition	Field Values	Comment
1	Message Type	0x23	
2	Header Version	0=MIL-STD-2045-47001 1=MIL-STD-2045-47001B	binary (0-16)
3 to 4	Status Tracking	1 to 65535	MSB-LSB
5	Connection type	Binary (1 to 255)	Refer to table 3.2.2.2.1-2
6 to 8	Reserved	constant = 0	for word alignment
The following reference message entry bytes [flagged by *] are repeated for R = 0 through 15 and will be appended after the reserved field above			
* 9 + (Rx8)	Status Code	0x00 to 0xFF	Refer to table 3.2.2.2.6-2
* 10 + (Rx8)	Group Code ID	0 = No group 1 = group 1 3 = group 3 4 = group 4 5 = group 5 6 = group 6	binary

Byte	Field Definition	Field Values	Comment
* 11 + (Rx8)	Repeat Code	0 = No repeat group 1 = repeat group 1 2 = repeat group 2 3 = repeat group 3	
* 12 + (Rx8)	repeat count 1 indicator	0 to 255	binary
* 13 + (Rx8)	repeat count 2 indicator	0 to 255	binary
* 14 + (Rx8)	field position indicator	0 to 255	binary
* 15 + (Rx8) to 16 + (Rx8)	Reserved	constant = 0	for word alignment

Table 3.2.2.2.6-2 VMF Status Codes

Status Code	Definition
If an error occurs during transmission of a message	
0x00 to 0x25	Refer to table 3.2.2.2.3-2
If an error is found in the message header by TCOMM before the message is sent	
0x40	Invalid Field
0x41	Error Orig Msg Format
0x42	Error ACK Msg Format
0x43	Error File Msg Format
0x44	Error Forward Msg Format
0x45	Invalid GPI
0x46	Invalid Bit Unit Name
0x47	Invalid Bit URN
0x48	Invalid Char Unit Name
0x49	Invalid Char URN
0x4A	Invalid Info Address Group
0x4B	Invalid Msg Size
0x4C	Invalid CANTCO RC
0x4D	Invalid CANTPRO RC
0x4E	Invalid Reply AMP

3.2.2.3 Communication with non-DII COE Compliant Systems

The TCS software segment will have to implement the message protocol expected by the non-compliant system because these messages will be passed through the DII COE Comm Server to the Tactical Communications Interface Software (TCIS) which will place the message into an “envelope” which will be sent to the TCIM for forwarding to the destination non-compliant system. Figure 3.2.2.3-1 shows a data flow diagram for this procedure.

The Generic Message Header - TCIS Message is defined in the *Interface Design Document (IDD) for the Tactical Communications Interface Module (TCIM) Common Interface Software (TCIS) to Applications Software, Rev Q,* dated May 15, 1997. This IDD was developed by Litton Data Systems, San Diego, Ca 92121.

ASAS and AFATDS are DII COE compliant systems, and there are no current plans for TCS to interface with any non-compliant systems through the TCIM. This section is included for completeness.

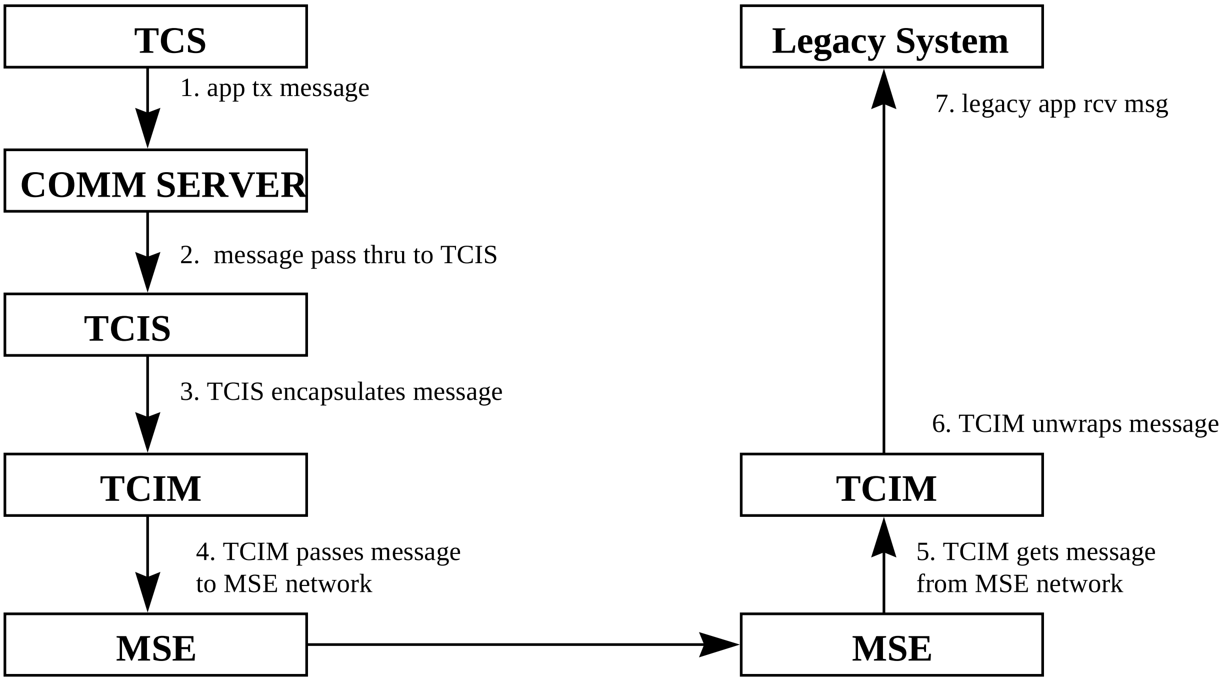


Figure 3.2.2.3-1. Message Transmission to a Non-DII COE Compliant Legacy System

3.2.3 Individual Data Element Characteristics

(Not applicable to these interfaces.)

3.2.4 Data Element Assembly Characteristics

(Not applicable to these interfaces.)

3.2.5 Communication Methods Characteristics

Communication between the TCS and the TCIM will be handled by the DII COE Comm Server Segment of the DII COE Kernel via the API implemented in the TCS software as described in Appendix A..

3.2.6 Protocol Characteristics

(Not applicable to these interfaces.)

3.2.7 Other Characteristics

(Not applicable to these interfaces.)

4. Requirements Traceability and Qualification Provisions

This section defines the requirements traceability between this document and the Operational Requirements Document it also defines the qualification methods to be used to ensure that each requirement of this interface has been met. Table 4.0-1 below traces the TCS to TCIM interface requirements to one or more of the TCS ORD requirements and the qualification method to be used to verify each requirement. Further, the TCS ORD requirements traceability to the TCS SSS, document number TCS-102, has been performed by the TCS Requirements, Analysis, and Design (RaD) Integrated Product Team (IPT). The results of this TCS ORD to TCS SSS requirements traceability are contained in Appendix B of the TCS SSS.

The qualification methods include:

D Demonstration The operation of the interfacing entities that relies on observable functional operation not requiring the use of instrumentation, special test equipment or subsequent analysis.

T Test The operation of the interfacing entities using instrumentation or special test equipment to collect data for later analysis.

A Analysis The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpretation, or extrapolation of test results.

I Inspection The visual examination of code, documentation, etc.

S Special Any special qualification methods such as special tools, techniques, procedures, facilities and acceptance limits.

Table 4-1 identifies the qualification methods to test the TCS to TCIM interface.

Table 4-1. TCS/TCIM Interface – Requirements and Qualification Methods

Function	C^4I System (Version)	ORD Requirement Number	Qualification Method
1. Initialize TCIM a. MSE connection b. SINCGARS connection	TCS/DII COE v3.1 for Sun Solaris 2.5	ORD016, ORD069 ORD070, ORD072, ORD109, ORD110	D, A D, A
2. Exchange messages with DII COE compliant		ORD016, ORD069	

Function	C^4I System (Version)	ORD Requirement Number	Qualification Method
system a. ASAS via MSE connection b. AFATDS via MSE connection c. AFATDS via SINCGARS connection	TCS/DII COE v3.1 for Sun Solaris 2.5	ORD070, ORD072, ORD109, ORD110	D, A D, A D, A
3. Exchange messages with non-DII COE compliant system	Not Applicable at this time.		

5. Notes

5.1 Background Information

The TCIM has the functionality to interface between the TCS host computer and the Army's MSE and SINCGARS communications equipment used to support Army Command and Control (C2) for deployed forces.

TCS will use the TCIM to permit the exchange of USMTF and VMF messages between the TCS and ASAS and AFATDS. The IDD's for ASAS and AFATDS identify the exact messages to be exchanged.

The TCIM is available in three versions : external chassis (compatible with most computers); internal, full-length PC/AT card for IBM PC compatibles; and standard Type II PCMCIA.

5.1.1 TCIM

5.1.1.1 TCIM Model 1 Description

The ruggedized TCIM supports critical operational communication interface requirements.

- a powerful, single-board specified by the U.S. Army Common Hardware Software (CHS) program, which consists of a microcontroller and two, high-performance Digital Signal Processors (DSPs)
- a mature interface architecture provides flexibility and growth capacity essential to support critical operational tactical communication requirements of joint service programs
- Software that allows the integration of host-based, local, and wide-area networks (LAN/WAN) into the tactical communications world
- communication channels designed to support GOSIP, OSI, DoD, and custom protocols
- The ability to pass data via wireline at rates of 64 kbps at distances up to 4 km
- each TCIM support two, programmable channels of communications
- up to seven TCIMs can be attached to each SCSI bus, providing up to 14 communications
- PC/AT, internal and external configurations
- SCSI I data connection

Table 5.1.1.1-1 shows the TCIM 1 Characteristics

Table 5.1.1.1-1 TCIM 1 Characteristics

FUNCTIONAL	
Processor	Microcontroller: 16 Bit Motorola MC68302 Digital Signal Processors: Two 24-Bit DSP56001s
Interfaces	Communications Equipment: Wireline, multipurpose, and two radio communication ports SCSI bus: Host data interface and SCSI bus extension (ANSI X3.131-1986)
Memory	Microcontroller: 768KB RAM and 256KB EPROM Digital Signal Processors: 288KB RAM channel 1, 192KB RAM channel 2
Reliability	Internal: 11,000 hours MTBF External: 9,333 hours MTBF
Maintainability	Internal/ External predicted MTTR of 0.25 hour
PHYSICAL	
Weight	External: 3.8 pounds Internal: 0.75 pound
Dimensions	External: 8" W, 1.6" D, 16" H Internal: Full PC/AT Card
ELECTRICAL	
Input Voltage	External: 18 to 36 Vdc (power supply required) Internal: + and - 5 Vdc
Consumption	External: 15 watts maximum Internal: 12 watts maximum
ENVIRONMENTAL	
Temperature	Operating: -25 to 49 degrees C Storage: -32 to 65 degrees C
Shock	30 degree rotational and transit drops per MIL-STD-810E, Method 516.4, Proc. IV & VI
Vibration	Tracked vehicle operation per MIL-STD-810E, Method 514.4, Proc. I, Category 8 and Proc. III, Loose Cargo
Altitude	Sea Level to 10,000 feet

Rain	1.8 inches per hour in 20 mph wind for 30 minutes
Humidity	Operating Range: 10% to 95% RH Non-operating range: 5% to 95% RH
Sand/Dust	20 mph for 30 minutes
Climate	Fungus resistant
Immersion	3 feet of water for 30 minutes in transit case
Explosive Atmosphere	No ignition
EMI	Conforms to the levels of FCC part 15, Class B

5.1.1.2 TCIM Model 2 Description

The TCIM 2 provides enhanced speed and options.

- has features of TCIM 1 with an upgraded microcontroller and high-performance Digital Signal Processor (DSPs)-the Integrated Multiprotocol Processor provides up to three times the processing power the TCIM 1 and the DSPs provide up to two times the processing power of the TCIM 1 DSPs
- upgraded to support two channels of X.25 communications and/or CMOS interface-based communications devices
- TCIM Common Interface Software supports TCIM 2 channel functionality and is backward compatible with the TCIM 1
- can support data via wireline at rates of 512 bps at distances up to 4 km
- supports interface to joint service tactical communications devices, such as JTIDs, at rates up to 2M bps

Table 5.1.1.2-1 shows the TCIM 2 Characteristics

Table 5.1.1.2-1 TCIM 2 Characteristics

FUNCTIONAL	
Processor	Microcontroller: 32 Bit Motorola MC68360 Digital Signal Processors: Two 24-Bit DSP56002s
Interfaces	Communications Equipment: Wireline, multipurpose, and two radio communication ports SCSI bus: Host data interface and SCSI bus extension (ANSI X3.131-1986)
Memory	Microcontroller: 2MB RAM and 256KB EPROM

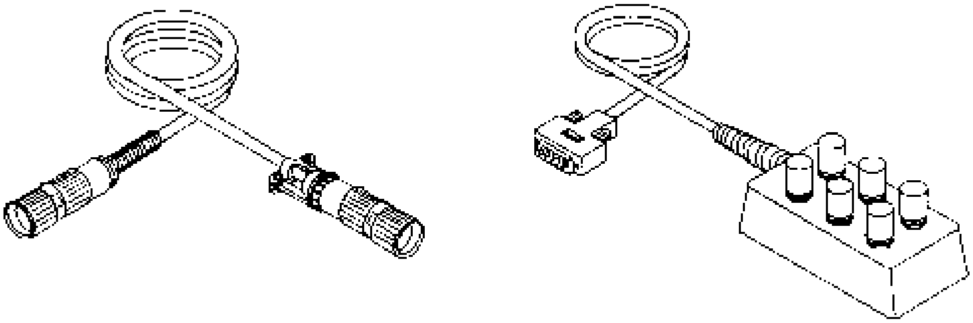
	Digital Signal Processors: 384KB RAM channel 1, 384KB RAM channel 2
Reliability	Internal: 11,000 hours MTBF External: 9,333 hours MTBF
Maintainability	Internal/ External predicted MTTR of 0.25 hour
PHYSICAL	
Weight	External: 3.8 pounds Internal: 0.75 pound
Dimensions	External: 8" W, 1.6" D, 16" H Internal: Full PC/AT Card
ELECTRICAL	
Input Voltage	External: 18 to 36 Vdc (power supply required) Internal: + and - 5 Vdc
Consumption	External: 15 watts maximum Internal: 12 watts maximum
ENVIRONMENTAL	
Temperature	Operating: -25 to 49 degrees C Storage: -32 to 65 degrees C
Shock	30 degree rotational and transit drops per MIL-STD-810E, Method 516.4, Proc. IV & VI
Vibration	Tracked vehicle operation per MIL-STD-810E, Method 514.4, Proc. I, Category 8 and Proc. III, Loose Cargo
Altitude	Sea Level to 10,000 feet
Rain	1.8 inches per hour in 20 mph wind for 30 minutes
Humidity	Operating Range: 10% to 95% RH Non-operating range: 5% to 95% RH
Sand/Dust	20 mph for 30 minutes
Climate	Fungus resistant
Immersion	3 feet of water for 30 minutes in transit case
Explosive Atmosphere	No ignition
EMI	Conforms to the levels of FCC part 15, Class B

5.1.1.3 TCS and TCIM Interfaces

TCS connects to the TCIM via a cable connecting the Small Computer System Interface (SCSI) ports on the TCS host computer and the external TCIM chassis.

TCIM uses the Wire Line Adapter shown in figure 5.1.1.3-1 to interface with MSE. TCIM uses the SINCGARS/KY-57 cable shown in figure 5.1.1.3-1 to interface with SINCGARS.

An external TCIM -II modem is shown in Figure 5.1.1.3-2.



SINCGARS/KY-57 Wire Line Adapter

Figure 5.1.1.3-1. TCIM Interface Cables

Figure 5.1.1.3-2. TCIM II

5.1.1.4 TCIM Supported Protocols

Table 5.1.1.4-1 lists the TCIM protocols included in the DII COE Comm Server.

Table 5.1.1.4-2 lists the protocols being developed for the DII COE Comm Server.

Table 5.1.1.4-3 lists the protocols that have been proposed for addition to the DII COE Comm Server.

Table 5.1.1.4-1. TCIM Protocols currently included in the DII COE Comm Server

MIL-STD 188-220 (JVMF)	Network Radio Protocol (NRP)
Defense Data Network (DDN)	Tank Network Radio Protocol (NRPT)
Simple Modem CNR	Network Radio Protocol for PRC-139 (NRPAWE)
Simple Modem GDU	Character-Oriented Message Catalog
FSAC Fire Support (FS)	Marine Tactical Systems Switched (ULS)
IFSAS Fire Support (IFS)	Marine Tactical Systems Broadcast (MTS)
IFS Smart (IFSS)	Marine Tactical Systems (MTS Version 5)
Gun Direction Unit (GDU)	Air Defense Interface (ADI/TADIL-B)
SINCGARS Control/Monitor (SCM)	FAAD Data Link (FDL/SINCGARS 2-2-2)
Maneuver Control System (MCS-11)	AFATDS Fire Support (AFS)
Army Data Distribution System (ADDS)	AFATDS Variable Message Format (VMF)

Table 5.1.1.4-2. TCIM Protocols being developed for the DII COE Comm Server

MIL-STD 188-220A	TADIL-J
Simple Phone Protocol (SPP/STU III/DSVT/DNVT)	

Table 5.1.1.4-3. TCIM Protocols proposed for the DII COE Comm Server

MCS-11 Circuit Switch (CSW)	Transmission Unit (TU)
ATDL	TADIL-A

5.1.2 DII COE

The DII COE concept is best described as an **architecture** that is fully compliant with the *DOD Technical Architecture for Information Management (TAFIM), Volume 3 - Architecture Concepts and Design Guidance,* an **approach** for building interoperable systems, a **reference implementation** containing a collection of **reusable software** components, a **software infrastructure** for supporting mission-area applications, and a set of **guidelines, standards, and specifications**. The guidelines, standards, and specifications describe how to reuse existing software and how to properly build new software so that integration is seamless and, to a large extent, automated. The DII COE will evolve as necessary to become compliant with emerging specifications, such as the *Joint Technical Architecture (JTA)* and *TAFIM Volume 7 - Adopted Information Technology Standards (ITS).*

The present DII COE reference implementation contains software written in C as well as Ada. However, the DII COE itself is concerned with the executable environment and is specifically designed to be programming-language neutral. It does not state a preference of one language over another, but leaves the selection of a programming language to higher-level standards profile guidance and programmatic considerations. When a selection is to be made, C++ is recommended over C while Ada95 is recommended over any earlier versions.

The DII COE is a “plug and play” open architecture. The current reference implementation is designed around a client/server model. The DII COE is *not* a system; it is a *foundation* for building an open system. Functionality is easily added to or removed from the target system in small manageable units, called *segments*. Structuring the software into segments is a powerful concept that allows considerable flexibility in configuring the system to meet specific mission needs or to minimize hardware requirements for an operational site. Site personnel perform field updates by replacing affected segments through use of a simple, consistent, graphically oriented user interface.

Figure 5.1.2-1 shows the DII COE Architecture. Refer to the *Defense Information Infrastructure (DII) Common Operating Environment (COE) Integration and Runtime Specification (I&RTS), Version 3.0, January 1997* for a complete description of the DII COE. This document can be downloaded from the Defense Information Systems Agency (DISA) Center for Standards worldwide web page at uniform resource locator (URL) http://spider.osfl.disa.mil/cm/general.html.

TCS will reside in the Mission Application layer and interface with the DII COE through the Standard Application Program Interfaces.

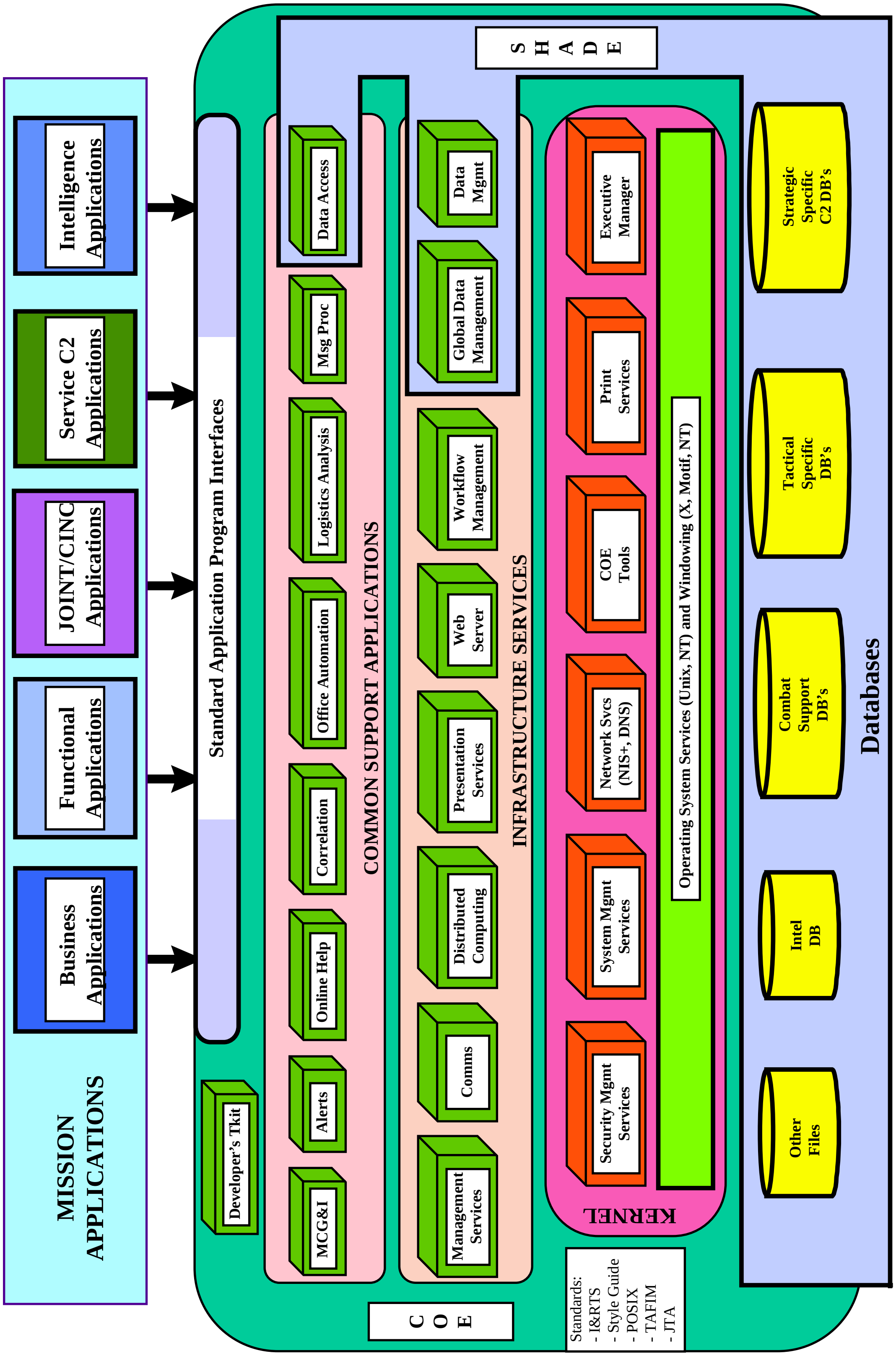


Figure 5.1.2-1. DII COE Architecture

5.1.3 MSE

MSE is a totally integrated communications system that provides the tactical U.S. Army commander with a secure, automatic, highly mobile, quickly deployable, survivable, tactical communications system capable of passing data, facsimile, and voice traffic throughout an entire five-division corps area of operations spanning 37,500 Square Kilometers, an area about the size of Massachusetts, Connecticut, and Rhode Island. These communications are essential for operations requiring swift, frequent movement, and rapid concentration and disbursement of forces, in a hostile combat environment.

The major items of equipment are integrated into five functional areas:

- Subscriber Terminals,
- Mobile Subscriber Access Radio Telephone Terminals,
- Wire Subscriber Access,
- Secure Automatic Switching, and
- System Control.

Subscriber Terminals provide the voice and data elements to interface with other functional areas of the MSE system. Mobile Subscriber Access radiotelephone terminals permit mobile and stationary users to automatically communicate secure voice and data throughout the tactical area of operations. Wire Subscriber Access allows non-radio users entry to the MSE system through concentrations of automatic switching equipment. Area coverage of the battlefield from mobile or fixed locations is achieved through secure automatic switching, continuous coverage, and the ability of commanders and staff to retain the same telephone number regardless of location. System Control provides an automated Corps-wide MSE system management capability, which is itself mobile, moving with the elements it controls.

MSE equipment is mounted in compact S-250 Shelters on High Mobility Multipurpose Wheeled Vehicles (HMMWV, or "Hummers"), that are easily transported on C-141, C5, C17, or C-131 aircraft; ready for rapid deployment. MSE also supports rapid relocation of command posts because all individual MSE assemblages can be set up or torn down within thirty minutes.

5.1.4 SINCGARS

The SINCGARS, shown in figure 5.1.4-1, provides commanders with a reliable, easily maintained Combat Net Radio (CNR) for command and control and provides Electronic Counter-Countermeasures (ECCM) against threat Electronic Warfare (EW).

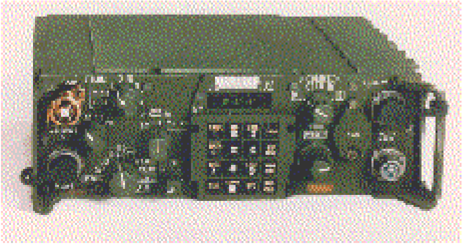


Figure 5.1.4-1. SINCGARS

SINCGARS configurations include: man-pack, vehicular (both low and high power), and airborne models. Communication Security (COMSEC) is integrated in currently produced versions of the ground and airborne models. The SINCGARS specifications are listed in table 5.1.4-1.

Table 5.1.4-1 SINCGARS Specifications

Frequency:	30 to 87.975 Mhz
Steps:	25 kHz
Channels:	2320
Offsets:	+/-5 and +/-10 kHz Single Channel
Presets:	8 Single Channels, 6 FH Channels/Nets
Modes:	FM Voice, AD1, AD2, TACFIRE, FSK Digital Data 600, 1200, 2400, 4800, 9600, and 16,000 bps Fixed and Frequency Hopping, Plain Text, Cipher Text Encryption.
TOD:	+/-10 PPM Accuracy
Night Vision:	Gen II Compatible
BIT:	Background, Power-up, and Extensive Operator Diagnostics
Power (manpack):	13 Volt (10.5V to 15V, Lithium BA-5590, Rechargeable Ni-Cad BB-590
Power (HUB):	Lithium BA-5372/U
Vehicular:	+24 Volts (22 to 32V)
Consumption:	Radio: 433 Milliamps average HUB 110 Microamps average
EMI:	MIL-STD-461A, MIL-STD-462A
TEMPEST:	Certified
Temperature:	-51C to 71C (-60F to 160F)
Immersion:	3 feet
Reliability:	>7000 hours typical
Maintainability:	Failure isolation to single circuit card without additional equipment
Weight:	6.9Kg (15.5 lbs)
Size:	W 27.8 cm (10.99 in.), D 25.4 cm (10.00 in.), H 8.7cm (3.42 in.)
Remote Control:	Full 2 Wire FSK Remote Control (10.99 in.), 2880 Hz (Control), 40 kHz (Voice)
Remote Fill:	Over-the-air rekeying
COMSEC:	Internal Encryption /Decryption, NSA approved
ECCM:	Internal Frequency Hop Module
Transmit Output:	Manpack: 0.5mw, 160 mw, 4W Vehicular: 50W

Transmit Noise floor:	<= -181 dBc/Hz at +/-10 Mhz from 30, 50, 75 Mhz
Transmit Harmonics:	<= -47 dBc for <= 3rd harmonics <= -75 dBc for >= 3rd harmonics
Receive Sensitivity:	-116 dBM for 10 dB SINAD -115 dBM for FH. 10% BER
Receive Selectivity:	For Undesired above ref., +/-100kHz >= 78 dB +/-200 kHz >= 90 dB +/-1.0 Mhz >= 113 dB +/-5 Mhz >= 127 dB +/-1.0 Mhz >= 141 dB
Distortion:	< 4%
Lo Radiation:	<= 24 dB uV
Squelch:	U.S. and NATO STANAG
Spurious Response:	<= 85 dB

5.2 Acronyms and Abbreviations

Acronym	Acronym and Abbreviations Definitions
A	Analysis
ABCS	Army Battle Command System
ACK	positive acknowledgment packet
ACS	Aerial Common Sensor
ADDS	Army Data Distribution System
ADI	Air Defense Interface
ADT	Air Data Terminal
AFATDS	Advanced Field Artillery Tactical Data System
AFMSS	Air Force Mission Support System
AFS	AFATDS Fire Support
AMPS	Army Mission Planning System
ARP	Address Resolution Protocol
API	Application Program Interface
ASD	Assistant Secretary of Defense
ASAS	All Source Analysis System
ASCII	American Standard Code for Information Interchange
ATCCS	Army Tactical Command and Control System
ATDL	Army Tactical Data Link
ATHS	Automated Target Handoff System
ATWCS	Advanced Tomahawk Weapons Control System
BER	Bit Error Rate
C^2	Command and Control
C^4I	Command, Control, Communication, Computers, and Intelligence
C^4ISR	C^4I Systems - Reconnaissance
CARS	Contingency Airborne Reconnaissance System
CANTCO	Can’t Communicate
CANTPRO	Can’t Process

Acronym	Acronym and Abbreviations Definitions
CCTV	Closed Circuit Television
CDR	(Artillery Target Intelligence:) Coordinate Report
CGS	Common Ground Station
CIGSS	Common Imagery Ground/Surface System
CJCSM	Commander Joint Chiefs of Staff Memorandum
CNR	Combat Net Radio
COMSEC	Communication Security
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
COE	Common Operating Environment
Comm	Communication
COMPASS	Common Operational Modeling, Planning, and Simulation System
COP	Common Operational Picture
COSIP	Computer Open Systems Implementation Program
CS	Communications Software
CSC	Computer Software Component
CSW	Circuit Switch
dB	decibels
DCGS	Distributed Common Ground System
DDN	Defense Data Network
DID	Data Item Description
DII	Defense Information Infrastructure
DISA	Defense Information Systems Agency
DL	Data Link
DNVT	Digital Non-secure Voice Terminal
DoD	Department of Defense
DoDI	Department of Defense Instruction
DSVT	Digital Secure Voice Terminal
DTG	Date Time Group
ECCM	Electronic Counter-Counter Measures
ETRAC	Enhanced Tactical Radar Correlator
EW	Electronic Warfare
FAAD	Forward Area Air Defense
FDL	FAAD Data Link
FH	Frequency Hop
FM	Frequency Modulation
FS	Fire Support
FSAC	Fire Support Ada Conversion
FSK	Frequency Shift Key
FTP	File Transfer Protocol
GDU	Gun Direction Unit
GOTS	Government Off-The-Shelf
GSM	Ground Station Module
GUI	Graphical User Interface
HAE	High Altitude Endurance

Acronym	Acronym and Abbreviations Definitions
HCI	Human Computer Interface
HDBK	Handbook
HMMWV	High Mobility Multipurpose Wheeled Vehicle
HP	Hewlett-Packard
HP-UX	Hewlett-Packard Unix
I	Inspection
IAS	Intelligence Analysis System
IDD	Interface Design Description
IEWCOMCAT	Intelligence and Electronic Warfare Character-Oriented Message Catalog
IFS	Initial Fire Support
IFSAS	Initial Fire Support Automate System
IFSS	IFS Smart
IPF	Integrated Processing Facility
IRS	Interface Requirements Specification
IP	Internet Protocol
IPT	Integrated Product Team
ITS	Information Technology Standards
I&RTS	Integration and Runtime Specification
JDISS	Joint Deployable Intelligence Support System
JIEO	Joint Interoperability & Engineering Organization
JMCIS	Joint Maritime Command Information System
JMF	Joint Message Format
JPO	Joint Program Office
JROC	Joint Requirements Oversight Council
JROCM	JROC Memorandum
JSIPS-AF	Joint Service Imagery Processing System - Air Force
JSIPS-N	Joint Service Imagery Processing System - Navy
JSTARS	Joint Surveillance Target Attack Radar System
JTA	Joint Technical Architecture
JVMF	Joint Variable Message Format (MIL-STD 188-220)
kHz	kiloHertz
LAN	Local Area Network
LCU	Lightweight Computer Unit
LRIP	Low Rate Initial Production
LSB	Least Significant Byte
MAE	Medium Altitude Endurance
Mbps	Mega bits per second (1024 * 1024 bits per second)
MCS	Maneuver Control System
MHz	MegaHertz (10^6 Hertz)
MIES	Modernized Imagery Exploitation System
MIL	Military
MSB	Most Significant Byte
MSE	Mobile Subscriber Equipment
Msgs.	Messages

Acronym	Acronym and Abbreviations Definitions
MTS	Marine Tactical System Broadcast Protocol (CNR) V4, V5 Mode 7 (Switched) (a.k.a. ULS, ULMS, and MTS Switched)
NAK	negative acknowledgment packet
NATO	North Atlantic Treaty Organization
NSA	National Security Agency
NITF	National Imagery Transmission Format
NRP	Network Radio Protocol
NRPTAWE	Network Radio Protocol
NRPT	Tank Network Radio Protocol
NSFS	Naval Surface Fire Support
NSWCDD	Naval Surface Warfare Center Dahlgren Division
ORD	Operational Requirements Document
OS	Operating System
PYLD	Payload
RaD	Requirements, Analysis, and Design
RC	Reason Code
RPC	Remote Procedure Calls
S	Special
SCM	SINCGARS Control Module
SCSI	Small Computer System Interface
SINAD	Signal Including Noise And Distortion
SINCGARS	Single Channel Ground and Airborne Radio System
SMTP	Simple Mail Transport Protocol
SNDCF	Subnetwork Dependent Convergence Function
SPIRIT	Special Purpose Integrated Remote Intelligence Terminal
SPP	Simple Phone Protocol
SSS	System/Subsystem Specification
STANAG	Standardization Agreement
STD	Standard
STU	Secure Telephone Unit
T	Test
TACFIRE	Tactical Fire (direction system)
TACCOM	Tactical Communications
TADIL	tactical data link
TAFIM	Technical Architecture For Information Management
TAMPS	Tactical Aircraft Mission Planning System
TBD	To Be Determined
TBMCS	Theater Battle Management Core System
TCIM	Tactical Communication Interface Module
TCIS	Tactical Communication Interface Software
TCP	Transaction Control Protocol
TCS	Tactical Control System
TDP	Tactical Data Processor
TEG	Tactical Exploitation Group

Acronym	Acronym and Abbreviations Definitions
TES	Tactical Exploitation System
TIDP-TE	Technical Interface Design Plan - Test Edition
TLM	Telemetry
TU	Transmission Unit
TUAV	Tactical Unmanned Aerial Vehicle
TV	Television
UAV	Unmanned Aerial Vehicle
UDF	User Defined Field
UDP	User Datagram Protocol
ULS	see MTS
URL	Uniform Resource Locator
URN	Unit Reference Notation
USA	United States Army
USAF	United States Air Force
USMC	United States Marine Corps
USMTF	United States Message Text Format
USN	United States Navy
VMF	Variable Message Format
WAN	Wide Area Network

Appendix A

DII COE Communication Server

Operational Overview

A.1 Overview

The COE Communication Software (CS) provides reliable message delivery through TCIM's, Tactical Radio, LANs and Wide Area Networks (WANs). The COE CS is the host resident portion of the communications profile, and it runs on a Lightweight Computer Unit (LCU), Hewlett-Packard (HP) 9000/700 Series computers under the HP-UX OS or Sun SPARCstation computers under the Sun Solaris OS.

The Comm Server CSC of the COE CS provides message transmission/reception for both Distributed Computing Environment (DCE) and non-DCE network environments. Communication via a Local Area Network (LAN) or DCE network uses either the Transmission Control Protocol (TCP)/Internet Protocol (IP), the User Datagram Protocol (UDP), File Transfer Protocol (FTP) or Simple Mail Transfer Protocol (SMTP). The Comm Server provides automatic protocol conversion from a generic message format to any TCIM-supported protocol, and sends converted messages for communication through the TCIMs. The Comm Server also enables host computers (client) without TCIMs to send/receive messages through TCIMs attached to other host (Server) platforms in the COE CS network. It uses DCE Remote Procedure Calls (RPC) for DCE cell communications.

The Comm Server CSC of the COE CS is used to configure messages for all protocols. It converts messages for transmission and reception between applications. The Comm Server communicates with one or more application programs through an API.

A.2 Application Program Interface

The Comm Server provides an external interface for application programs in C and Ada for message passing and configuration/administration of the Comm Server. The API is through the **tcomm_read()** and **tcomm_write() library** function calls.

The **tcomm_read()** and **tcomm_write()** functions use shared memory and Unix message queues to pass up to 1.5 megabytes of data between the Comm Server and the application. The **tcomm_read()** and **tcomm_write()** functions are contained in the archive library **tcommlib.a**.

A.2.1 tcomm_write() description

The **tcomm_write()** function is defined as follows:

int tcomm_write(int qkey, long int len, unsigned char *message)

A.2.1.1 Input Parameters

qkey = queue key for interprocess communication
len = length of message in bytes
***message** = pointer to the message

A.2.1.2 Output Parameters

The **tcomm_write()** function returns an integer indicating completion status:

0 = success
1 = Failed to allocate shared memory
2 = Failed to attach shared memory to process
3 = Failed to write to queue

A.2.2 tcomm_read() description

The **tcomm_read()** function is defined as follows:

int **tcomm_read(int qkey, unsigned char **message, int WAIT_FLAG)**

A.2.2.1 Input Parameters

qkey = queue key for this processes input queue
****message** = pointer to a pointer to the message, **tcomm_read()** copies the message into the address specified by the pointer
WAIT_FLAG = Toggle that indicates whether **tcomm_read()** will suspend until there is a message to read. WAIT = 0; NO WAIT = 1

A.2.2.2 Output Parameters

The **tcomm_read()** function returns an integer indicating the number of bytes read (1 to 1,500,000). If the NO_WAIT flag is set, **tcomm_read()** will return 0 if there is nothing to read; otherwise it will suspend the process until there is data to read.

Additional information about the **tcomm_read()** and **tcomm_write()** functions is contained in section 3 of reference IDD/N250-92-L029-009.